

Programování v jazyce C pro chemiky (C2160)

2. Příkazy cyklu while a do

Příkaz cyklu `while`

- Příkaz cyklu **while** slouží k opakovanému provádění příkazů
- Zapisuje se:

```
while (podmínka)  
{  
    příkazy;  
}
```
- Příkazy se budou vykonávat opakovaně tak dlouho, dokud je splněna podmínka (ta se testuje před každým průchodem)

```
int a = 0, b = 3, c = 1;
```

```
while (a < 5) {  
    c = c * b;  
    a = a + 1;  
}
```

Příkaz cyklu do

- Cyklus **do** funguje podobně jako **while**, ale podmínka je uvedena na konci
- Zapisuje se:
do
{
 příkazy;
} **while** (*podmínka*);
- Na rozdíl od cyklu **while** při použití cyklu **do** vždy dojde minimálně k jednomu průchodu cyklem (podmínka se vyhodnocuje až po průchodu)
- Na konci cyklu je za podmínkou vždy středník

```
int a = 0, b = 3, c = 1;  
  
do {  
    c = c * b;  
    a = a + 1;  
} while (a < 5);    // Nezapomínat na středník na konci
```

Příkaz break

- Příkaz **break** slouží k vyskočení z cyklu
- Po vykonání příkazu **break** program pokračuje kódem za cyklem

```
int a = 0, b = 3, c = 1;

while (a < 5) {
    c = c * b;
    if (c > 100)
        break; // Zde se vyskoci z cyklu (pokud c > 100)
    a = a + 1;
}

a = b + c; // Tady program pokračuje pokud byl cyklus ukončen
           // nebo pokud z něj bylo vyskoceno pomoci break
```

Příkaz break

- V případě vnořených cyklů příkaz **break** ukončí jen ten nejvnitřnější, v němž se přímo nachází

```
int a = 0, b = 2, c = 1;

while (a < 5) {
    b = 2;
    while (b < 4) {
        c = c * b;
        if (c > 100)
            break; // Zde se vyskoci z vnitřního cyklu
        b = b + 1;
    }
    a = a + 1; /* Tento příkaz se provede celkem petkrát, *
               * vždy po skončení vnitřního cyklu *
               * (at už kvůli breaku nebo podmince) */
}
}
```

Dodržujte následující pravidla

- Na začátek programu nezapomeňte vložit hlavičkové soubory:
`#include <stdio.h>` (pokud používáte funkce `printf()` a `scanf()`)
`#include <math.h>` (používáte-li matematické funkce `sqrt()`, ...)
Při použití matematických funkcí je navíc potřeba použít parametr `lm` (l je zde malé L), aby došlo k připojení matematické knihovny
- Všechny proměnné vždy inicializujte vhodnou implicitní hodnotou při její definici
- Za definicí každé z proměnných uveďte krátký komentář popisující, k čemu proměnná slouží
- Nezapomeňte, že v podmínkách `if` a v podmínkách cyklů musíme při porovnávání proměnných používat dvě `==`, nikoliv jedno `=`.
- Ve funkci `scanf()` nezapomeňte před název proměnné přidat `&`. Toto však neplatí pro funkci `printf()`, kde se tento znak nepoužívá.
- Dbejte na správné odsazování textu. Při správně nastaveném editoru by mělo ke správnému odsazování textu docházet automaticky

Úlohy

1. Vytvořte program, který vypíše čísla od 1 po hodnotu zadanou uživatelem. **1 bod**
2. Vytvořte program, který si vyžádá od uživatele celé kladné číslo a určí, jestli se jedná o prvočíslo. **1 bod**
3. Vytvořte program, který vypíše prvních n prvočísel. Číslo n bude zadáno uživatelem na začátku. **nepovinná, 2 body**
4. Výše uvedený program upravte tak, aby vypisoval prvočísla oddělená čárkou, vždy deset prvočísel na každý řádek. **nepovinná, 1 bod**